Full-Scale Continuous Synthetic Sonar Data Generation with Markov Conditional Generative Adversarial Networks

Full-Scale Continuous Synthetic Sonar Data Generation with Markov Conditional Generative Adversarial Networks*

Marija Jegorova[1], Antti Ilari Karjalainen[2], Jose Vazquez[2], Timothy Hospedales[1]

***Abstract*— Deployment and operation of autonomous underwater vehicles is expensive and time-consuming. High-quality realistic sonar data simulation could be of benefit to multiple applications, including training of human operators for post-mission analysis, as well as tuning and validation of autonomous target recognition (ATR) systems for underwater vehicles. Producing realistic synthetic sonar imagery is a challenging problem as the model has to account for specific artefacts of real acoustic sensors, vehicle attitude, and a variety of environmental factors. We propose a novel method for generating realistic-looking sonar side-scans of full-length missions, called Markov Conditional pix2pix (MC-pix2pix). Quantitative assessment results confirm that the quality of the produced data is almost indistinguishable from real. Furthermore, we show that bootstrapping ATR systems with MC-pix2pix data can improve the performance. Synthetic data is generated 18 times faster than real acquisition speed, with full user control over the topography of the generated data.**

I. INTRODUCTION

In underwater environments, sonars are often preferred over other sensors due to the high density of organic material and inorganic dust that can restrain optical visibility. Because of their perceptual robustness, sonar sensor data is heavily relied upon for tasks such as object localization, oil-pipe and infrastructure inspections, search and rescue, and other commercial and military applications.

A vast amount of data is required to construct detection and recognition models for automating most of these applications. Underwater data collection is expensive, time-consuming, and in most cases commercially sensitive. A means of synthetically creating such data would be highly beneficial to the underwater sensor processing community, as it would mitigate the costly process of data collection by instead making better use of the available real training data.

Existing techniques for image synthesis, such as generative adversarial networks (GANs) [1] have recently grown capable of producing and enhancing images of high resolution (e.g. 2048×1024 by pix2pixHD [2]). However, typical underwater survey missions sonar images usually exceed the image resolution of 300,000×512 pixels. We propose the Markov Conditional pix2pix (MC-pix2pix) method which, to our knowledge, is the first method capable of generating realistic sensory output for full-length missions, given a small amount of initial training data. Crucially, such generation runs 18 times faster than acquisition on the real hardware, resulting in a realistic and faster than real-time simulator.

To demonstrate the utility of our approach, we provide quantitative results in two extrinsic evaluation tasks: (i) the synthetic data is almost impossible to distinguish from real data for domain experts, thus enabling training of teleoperators without using real hardware; (ii) significant performance gains are achieved when using this synthetic data to augment training datasets for autonomous target recognition (ATR) in a variety of seabed conditions. The results presented in this work are produced with Marine Sonic sonar side-scan data, but the method itself is sonar-agnostic.

II. RELATED WORK

GANs [1] are a class of neural network models for the realistic data generation. Since their initial introduction in 2014, a large number of extensions have been proposed for various applications, primarily focused on realistic image and video generation [3], [4], [5], [6], [7], where only a limited amount of training data is available. In contrast to these tasks, there is comparatively little work investigating how GANs can be of benefit in robotics. Although robots that use image recognition in domains where training data is scarce may benefit from the conventional applications of GANs. Some applications that are more relevant to robotics include GAN-based approaches to imitation learning [8], [9], which allow robots to efficiently learn a single policy or a discrete set of policies from demonstration, and direct generation of robot control policy repertoires [10], a technique that enables sampling from the continuous target-conditional distributions over the control policies within a scope of a given task.

Facilitation of the user-controlled simulation requires some form of the information transfer. GANs have been extensively used for style transfer and image-to-image translation, beginning with cycleGANs for transfer between unpaired images [11]. However, on paired image translation problems – the task we are primarily concerned with – pix2pix [12] and its subsequent variations [2], [13] are known to perform considerably better. No current image translation methods can be directly applied to full-mission sonar data because of the extremely high resolution. The size of the full image to be generated is usually in excess of 300,000×512 pixels – roughly the amount of data generated by a short two hour training mission. Our method solves this problem by producing such image in a piece-wise sequential manner, and ensures continuity of the output through the use of a Markov assumption. The use of Markov assumption here is justified by the temporal nature of the real data acquisition during real mission, as well as by the general spatio-temporal

* This work was supported by SeeByte Ltd

[1] University of Edinburgh, UK m.jegorova@ed.ac.uk, t.hospedales@ed.ac.uk

[2] SeeByte Ltd., UK antti.karjalainen@seebyte.com, jose.vazquez@seebyte.com

Fig. 1: **Pipeline:** 1-2. Training instructor labels the map regions with desired textures and target locations. 3. Trainee operator creates a route across a given map with an objective of collecting data for locating hidden targets. 4. The model receives semantic maps and route as inputs and outputs synthetic sonar data for the entire mission. (In real life vehicle completes the mission delivering the sonar data collected.) 5. Example of sonar images when inspected by a human operator.

continuity of the required data.

The previous attempt of generating sequential data with GANs, recurrent GANs (RGANs) and recurrent conditional GANs (RCGANs) [14], focused on medical sequence generation. This has been accomplished through the use of recurrent neural network architectures. There are two issues with applying these techniques to the problem of synthetic sonar imagery generation. Firstly, we require the control over the topography. So the model architecture would need to be modified for image translation with convolutional layers, which would further require one to use backpropagation through time for training the network, rendering the training process computationally intractable for the size of data that we work with. Secondly, RGAN and RCGAN are designed to produce semantically realistic sequences, whereas we require perceptually realistic image sequences. Additionally, the nature of the sonar imagery suggests that the Markov assumption alone is enough for the coherency and continuity.

A small number of papers address the underwater robot perception problems with GANs: the work of [15] shows cycleGANs enhancing synthetic target objects for embedding them into the real sonar images in order to train an ATR system, while [16] proposes a method for refining video images rather than generating new acoustic imagery.

Until now the applications of GANs to underwater sensory data were mostly enhancing the imagery, either optical or acoustic, rather than generating brand new data. MC-pix2pix is also the first model of its type addressing the generation of a whole mission's worth of data rather than smaller images.

III. PROBLEM AND MOTIVATION

A. Why generate sonar images?

The key application for high-quality simulation is bootstrapping autonomous target detection and recognition (ATR) methods when training data is scarce, or some types of seabeds are underrepresented in the real training set. In section VI-B we show improvements in the ATR performance when generating a variety of seabeds and introducing them into the ATR training together with the available real data.

Realistic simulation could also benefit the training of teleoperators for mission planning and interpretation of sonar imagery, replacing the costly real data collection.

B. Synthetic framework for training of the vehicle operators

The simulation pipeline, presented in the Figure 1, assumes that the training instructor marks the regions of the map with a specific topography, such as rocks, ripples, clutter, and objects of interest. The trainee operator is presented with this map without the target objects marked, and creates a route over it that should allow the robot to locate these hidden objects. Given semantic maps provided by an instructor and the route created by trainee operator, the purpose of our technique is to generate realistic seabed scans for the entire mission. Methods such as [15] can be employed to embed the target objects in the requested locations in the synthetic seabed scans, and can then be displayed to the trainee operator for visual inspection and object detection, just like during a real post-mission analysis.

The emphasis of this work is on keeping synthetic data maximally consistent and realistic, whilst achieving the highest generation speed possible.

C. Problem Specification

For the task described above the following requirements should be considered:

- Realistic looking synthetic data generation: the main focus of this work. Our method is based on GANs because they have been identified as the current best approach for generating realistic imagery.
- Spatial coherency: imagery of the entire mission should appear continuous and consistent. The paired nature of pix2pix guarantees consistency within topographical features represented as different labels in semantic maps. Additional conditions are introduced in section V, and improve the continuity of the output further.
- Viewpoints invariance: the same section of the map should appear texture-consistent when observed from different viewpoints.
- Speed of generation: in practice, for faster than real-time simulations, our model should be significantly faster at test time than real-time sonar data acquisition.

IV. EXPERIMENTAL SETUP

A. Training Data

The real side-scan sonar data used for the experiments is acquired with a Marine Sonic sonar. Its across-track resolution is 512 pixels ($\times 2$ for both port and starboard channels). The vehicle travels at the speed of 1 meter per second and generates approximately 16 pings per second. As the vehicle turns it causes distortions in the images, and models that generate synthetic imagery should be expected

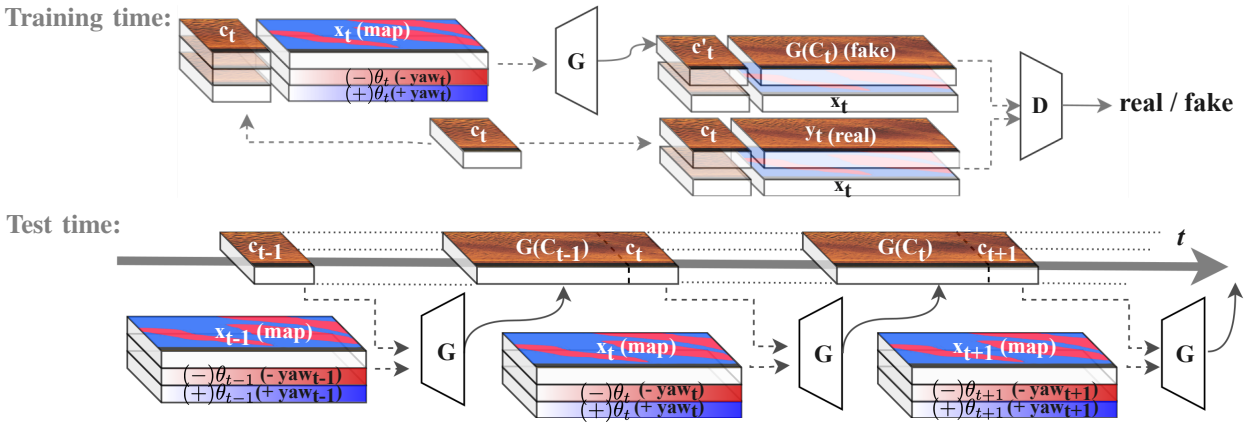

Fig. 2: **Training time:** similarly to the pix2pix, the generator inputs semantic maps corresponding to the desired topography x_t, outputs synthetic sonar-scan data $G_t(C_t)$. It is extended to accept two additional conditions - a snippet of the previous image c_t facilitating continuity in the generated mission and yaw indicating the requested turns of the vehicle. Output is then labelled by discriminator as real or fake along with the real images. **Test time:** at each time-step, the generator processes a semantic map of requested configuration, yaw variable (responsible for turn distortions, defined by the vehicle trajectory), and a small snippet of the previous synthetic image to enforce the continuity of the seabed throughout the mission.

to produce similar distortions. Our model accounts for these distortions using the desired vehicle attitude information (yaw, pitch, and roll). Only yaw information is provided by the gathered training data, but we note that our method is able to incorporate pitch and roll data as well.

To create a training dataset, sonar scans were sliced into 464×512 images. Our model was trained on a relatively small dataset of 540 of these images (and their corresponding semantic maps). Increasing the training set size might bring further improvements but in our experience this method works with as few as 200 training image samples.

B. Assessment metrics

In addition to the visual examples provided in Figure 3, the model performance has also been quantitatively assessed using the following metrics:

- Human visual assessment score: we provide the statistics on distinguishing real sonar imagery from the synthetic images. It is collected from 30 participants with a variety of experience of working with underwater sonar data. During the test, participants were allowed to inspect images without the time limit.
- Freschét Inception Distance [17]: often used for quality assessment of generated images, this is a heuristic for measuring the difference between the real and synthetic image distributions.
- Generation speed at test time: crucially for practical application, generative models must provide results of the requested quality without compromising the speed of generation. A minimal requirement is an order of magnitude faster than real data collection speed.
- Performance improvements of an ATR detection algorithm when bootstrapped with the synthetic data along with the available real data. This is assessed in terms of mean Average Precision (mAP) and F1-score - harmonic mean between the precision and the recall of the ATR system.

V. METHOD AND ARCHITECTURE

An overview of the model architecture is provided in Figure 2[1]. It resembles the fully-convolutional pix2pix architecture with 9 resnet blocks [12]. Importantly, it is designed to accept two conditions at the input level [18].

1) Conditions: the first condition, c_{t-1}, enforces the visual continuity of the generated output at test time. The information is conveyed by a short snippet taken from the end of the previous image, enabling the model to run self-conditionally at test time, as illustrated in Figure 2.

The second condition, a yaw-based metric θ_t, takes care of the image distortions caused by turns, it is calculated as:

$$\theta_t = 5 \ \ max(1, |\psi_t - \psi_{t+50}|) \tag{1}$$

where ψ_t is the yaw for ping t, and the sign of $(\psi_t - \psi_{t+50})$ is used to determine whether the clockwise or counterclockwise turn is expected. θ_t is calculated per ping (per row) of the corresponding semantic map x_t, the resulting vector gets repeated column-wise, separated into two arrays based on the sign of $(\psi_t - \psi_{t+50})$, and overlayed with the single-channel semantic map x_t, completing the generator input.

2) At training time: the generator inputs the single-channel semantic maps x_t and the two conditions - yaw variable θ_t and the previous image snippet c_t. The generator outputs single-channel generated sonar images $G(x_t, ...)$. the discriminator receives all available data except the yaw variable - semantic maps x_t, condition c_t, and real images y_t, and generated images $G(x_t, ...)$. The discriminator outputs the verdict on whether the image is real or fake. The discriminator is rewarded based on how well it can distinguish

[1]Please note: semantic maps, sonar-scans, and yaw variables are single-channel and are only coloured as RGB for illustration purposes.

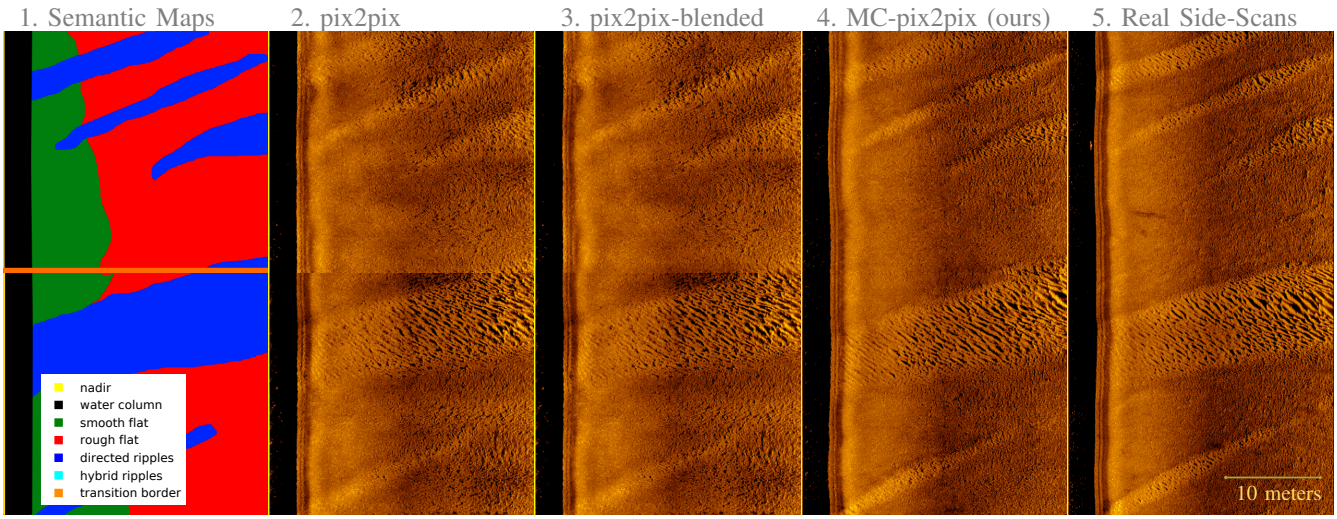

Fig. 3: **Visual Comparison (left-to-right): 1.** Semantic maps are used as an input by all of the compared models. In reality the semantic maps are grey-scale with different shades corresponding to different types of terrain. Colour is introduced for visualisation purposes only. Border-line label indicates the transition between the images for convenience of the reader and is not present in the input of the model. **2.** Original pix2pix example has a clear sharp transition border between the images (in the middle). This is because the image patterns or intensities are not shared between adjacent images. **3.** pix2pix with sigmoid-smoothing applied at the transition demonstrates that simple post-processing is not particularly good at matching the textures of the seabeds. **4.** MC-pix2pix (ours) has clearly smoother transition border, enabling it to produce continuous imagery for missions of any length when run repeatedly. **5.** The real data example.

the synthetic image $G(x_t, ...)$ from real y_t, generator - based on if it managed to "fool" the discriminator.

This model is adversarially trained for 200 epochs with batch-size 10 and 3 repetitions of discriminator for 1 of generator per each epoch with the following loss function:

$$\begin{aligned} G_t^\star = \arg\min_G \max_D \big\{ & \mathbb{E}_{x_t,y_t}[logD(x_t, y_t)] \\ + \ & \mathbb{E}_{x_t,\mathbf{C}_t,z}[1 - logD(x_t, G(x_t, \mathbf{C}_t, z))] \\ + \ & \mathbb{E}_{x_t,\mathbf{C}_t,y_t,z}[\|y_t - G(x_t, \mathbf{C}_t, z)\|_1] \big\} \end{aligned} \quad (2)$$

where x_t are semantic maps, y_t are real sonar images, z is random noise vector, and $\mathbf{C}_t = [c_{t-1}, \theta_t]$ is a collection of condition variables for the generator. The first two lines of (2) represent the discriminator and generator losses respectively, and the last one is the L1 loss, a regularization term that is meant to discourage blurring in the generator output [12].

3) At test time: only the generator is used and runs identically to the train-time, except c_t now comes from the end of the previous image generated. The model output is therefore dependent on its own previous output and capable of producing consistent and continuous images of any length.

VI. RESULTS AND COMPARISONS

A. Experiment 1: Image quality assessment results

In this experiment we compare MC-pix2pix with real images as well as with the output of the original pix2pix and pix2pix with post-processing, i.e., with blending the border-line between the separate synthetic snippets using sigmoid-function smoothing. The achieved results are compared both qualitatively and quantitatively as follows:

1) Visual examples: of all the methods are provided in Figure 3. These are directly comparable as they are generated from the same semantic maps (left), obtained via segmenting the real seabed images (right). Their underlying generative models are trained for the same number of epochs on the same dataset. In order to further eliminate the disadvantage for baseline methods that do not use the yaw variable, no yaw variation was applied in this example (i.e., no turns). This example is primarily illustrating the consistency of the MC-pix2pix output compared to the baselines.

2) Visual assessment scores: are obtained from 30 human experts (different levels of experience with sonar data - from introductory course to several years of work with sonar images). Although it is common to use Amazon Mechanical Turk for obtaining such assessment, it is not feasible in our study since real data are both commercially sensitive and too specialized to get a valuable assessment by people previously unexposed to the sonar imagery. Instead we obtain our assessments from the human experts who possess some knowledge of the domain.

The test consists of a number of images generated by MC-pix2pix, pix2pix, sigmoid-blended pix2pix, and corresponding real examples presented in even proportions for a human expert to classify as real or synthetic. The order of images from different models is randomised to avoid putting any of the methods into a disadvantage of being examined last.

Results are presented in Table I. Domain experts had 0.52 mean accuracy labelling MC-pix2pix images as real or fake. This is essentially an optimal result because for a two-class problem (real or fake), proximity to 0.5 means experts being as good as random at telling the synthetic data apart from

Metrics	pix2pix	sigma-pix2pix	MC-pix2pix
Mean accuracy of labeling	0.64	0.62	**0.52**
Synthetic labeled as real	0.34	0.42	**0.54**
Mean time per image (sec)	4.85	4.86	**6.13**
Freschét Inception Distance	0.9257	1.0241	**0.7834**

TABLE I: **Image Test Scores:** the average accuracy around 0.5 shows that humans are as good as random at telling MC-pix2pix images apart from real. MC-pix2pix gets labelled as real more than the competitors and comes the closest to the 0.66 ratio of real images labelled as real. Image processing times show MC-pix2pix images are the most challenging to inspect. The lowest FID score confirms the MC-pix2pix is closer to the real image distribution than the competitors.

Test: Flat	Real only	+ Noise	+ SonarSim	+ MC-pix2pix
mAP	0.30	0.39	0.27	**0.45**
F1-score	0.57	0.58	0.50	**0.60**
Test: Complex	Real only	+ Noise	+ SonarSim	+ MC-pix2pix
mAP	0.00	0.01	0.00	**0.11**
F1-score	0.23	0.59	0.62	**0.68**

TABLE II: **Bootstrapped ATR performance improvement**: MC-pix2pix improves ATR mAP and F1 score compared to just using real data, as well as beats the baselines for both non-complex flat (top) and complex (bottom) test terrains.

real. Further we present the proportion of synthetic data mislabelled as real (i.e., the success of generator in "fooling" human experts). For comparison, the proportion of the real data labelled as real is 0.66. The last metric of the visual assessment is the average time taken to make a decision on a sample. Interestingly, participants spend more time on MC-pix2pix images, which suggests these were more challenging to classify. Our method compares favourably to all of the presented baselines for all the presented metrics.

3) Freschét Inception Distance (FID): is also provided at the bottom of the Table I. Lower values correspond to the distributions closer to the real one. For instance the FID between two real data sets would be approximately zero, whereas the FID between a constant and $U(0,1)$ is greater than 6. FID is calculated on test images of size 1856×512. This size was chosen arbitrarily - similar results are expected from larger images. FID is sensitive to scaling, so the data for FID assessment has been normalized to the [0, 1] range.

4) Generation speed: the MC-pix2pix is expected to be fairly close to the original pix2pix in speed due to our model being an extension of pix2pix. We used GTX 1080 Ti (12GB RAM) for estimating the MC-pix2pix generation speed. MC-pix2pix is approximately 18 times faster at test time than the real acquisition speed. Marine Sonic acquires 17,100 $\pm$ 10% pixels per second depending on the settings of the sonar.

B. Experiment 2: Improving ATR training with MC-pix2pix

1) Motivation: training ATR on simulated data is useful in case of the lack of complexity in training data, or the lack of training data itself, in which case adding more realistic simulated data would be beneficial. If certain seabed types are underrepresented in the currently available training set, but MC-pix2pix was exposed to these types of terrains before, it can enrich the dataset with additional seabed types.

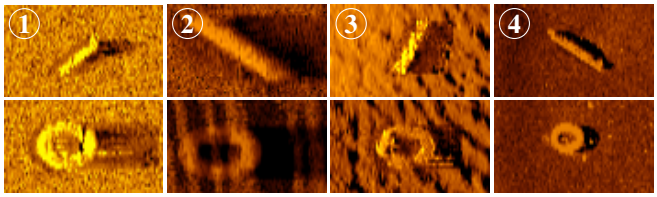

Fig. 4: **Examples of target objects (tyres and cylinders) for the ATR training:** 1. Random uniform noise background, 2. SonarSim[2], 3. MC-pix2pix, and 4. Real data. Objects in pictures 1-3 are synthetic, inserted with the [15][3] method.

2) Experiment goals and the ATR network: in both of these cases we check the increase in the ATR performance between training on just a small real dataset and enriching it with MC-pix2pix. We assess the performance with mAP and F1-score, as explained in Sec. IV-B. We are interested in the increase in the ATR performance only - the performance level itself is irrelevant here. The ATR method used in this test is a generic RetinaNet-type network [19].

3) Baselines explained: we train 4 ATR networks on the corresponding datasets: real data only (flat and non-complex), the same real data plus MC-pix2pix images, and baselines - real dataset plus uniform random noise backgrounds, and real dataset plus SonarSim seabeds[2]. All except the real data are augmented with synthetic targets using the method from [15][3], examples of these are provided in Fig. 4.

4) Experiment 2.1: Data shortage: for this experiment MC-pix2pix was trained on the available real training set (flat and non-complex). The MC-pix2pix-generated data were used to train the ATR, which then was tested on another flat and non-complex dataset. Table II (top) shows that MC-pix2pix provides significant improvements in MAP, compared to just real data and other baselines, and the best F1.

5) Experiment 2.2: Lack of complexity: in this case MC-pix2pix was pre-trained with slightly more complex ripply seabeds, emulating previous exposure to the complex data. It then generated more of the complex seabeds, that were used to train the ATR alongside the flat and non-complex real data. When testing this ATR on complex real terrains (results presented at the bottom of Table II), both F1-score and mAP drastically improve with the MC-pix2pix data bootstrapping, compared to just real data training and baselines.

This confirms that MC-pix2pix could be deployed as a highly efficient bootstrapping technique for improving ATR performance in cases of low real data availability or low real data diversity, that are common in the real life applications.

[2]SonarSim - standard vaguely realistic side-scan simulator as used in [15], capable of generating various seabed textures with limited user control over the type of generated data, but not the exact topography.

[3]Due to extremely low amount of the training data for targets we could not generate targets with MC-pix2pix.

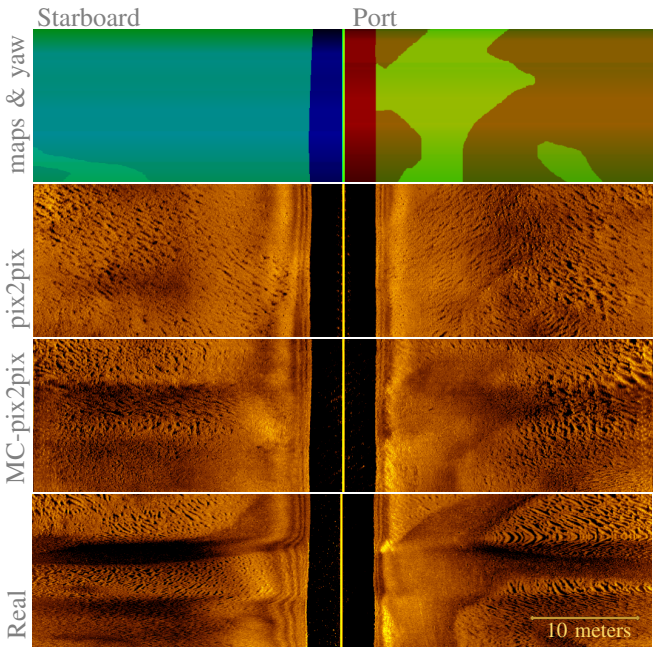

Fig. 5: **Reproducing turns with yaw-conditioning (top-to-bottom):** 0. semantic map overlayed with yaw. 1. pix2pix fails to capture a distortion caused by vehicle turning as semantic map provides no indication of turns other than perhaps indirectly through the topography. 2. MC-pix2pix benefits from a inbuilt yaw-based condition, getting close to real turn patterns and distinguishing between the inside (left) and the outside (right) turns. 3. Real example.

C. Addressing Potential Concerns:

1) Advantages compared to generating the seabed piece-wise and then stitching it together with post-processing: Although standard smoothing techniques like sigmoid-smoothing interpolate well between the colours, they do not conduct the texture integration between the images as is evident from the Fig. 3 middle section (at border-line).

2) Quality Decay over the generation time: self-conditional model at test time suggests that the quality drop could accumulate over time. However, the nature of GAN architecture prevents this - trained GAN always samples the training data distribution, regardless of the condition. This has been verified via generating a standard test mission - average duration of 2 hours, 300,000 pixels along the track.

3) Handling vehicle turns: we condition on the yaw only (because roll and pitch are not available in our data). It does not seem to be quite enough information to make results fully realistic, however the model not only acknowledges the concept of a turn distortion but also distinguishes between the inside and the outside turns, in some cases producing very realistic results (especially successful for the outside turns simulation). A visual example of what real vs. generated turns look like is presented in Fig. 5.

4) Handling multiple viewpoints: typically a vehicle observes the same area at least twice. The MC-pix2pix accounts for the topographical coherence with respect to the terrain types. The model naturally produces a similar image for the different viewpoints. Example in Figure 6 shows two views of the same area (e.g. the image synthesized for the same map approached from different sides) and their overlay.

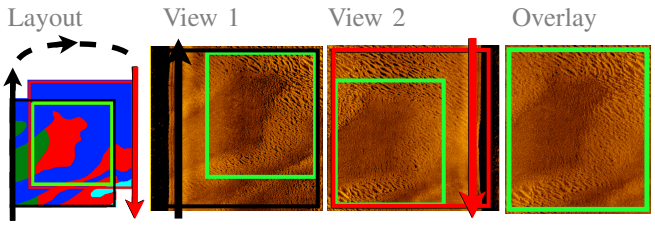

Fig. 6: **Handling different view-points:** MC-pix2pix is topographically consistent with respect to the types of terrains. This example shows the same region, generated as perceived from two different viewpoints with some displacement. Generated images show a clean overlap without contradictions.

5) Unconditional generation: this work focuses on producing the missions with user-controlled topography, however if one needs to avoid specifying it (e.g. when producing the training data for ATR) the original GAN [1] and modifications, such as DCGANs [20], can be employed to generate some semantic maps for the input into MC-pix2pix.

6) Other potential applications: MC-pix2pix can be potentially used in any setting where simulation of large-scale continuous data is required. Useful for bootstrapping learning algorithms, environment feature detection and recognition, or even for side-scrolling games background generation.

VII. CONCLUSIONS AND FUTURE WORK

This work proposes a method for generating realistic synthetic sonar sensory data for full-length underwater missions with a direct control over the topography. To our knowledge this is the first published work that addresses generation of side-scans for entire missions with generative adversarial networks. Examples of visual results were provided along with the quantitative assessment of the model. These include FID scores, generation speed, ATR performance improvement results when bootstrapped with the MC-pix2pix generated data, and visual assessment scores confirming MC-pix2pix synthetic data looks very realistic to humans.

In future work we will investigate the use of roll and pitch data for improving quality of the simulation for the vehicle turns, subject to the availability of the suitable training data.

The main extension to this work is generation of the data for higher fidelity sonars, such as EdgeTech (an order of magnitude higher resolution compared to the Marine Sonic data presented in this work), or SAS sonars (two orders of magnitude higher in resolution compared to Marine Sonic). Despite being very challenging this problem can be addressed with some limited extensions to the current MC-pix2pix algorithm and is currently a work in progress.

ACKNOWLEDGMENT

We thank Stephanos Loizou and Peter Scanlon for their help with the ATR experiments, and Roshenac Mitchell, Joshua Smith, and Henry Gouk - for the technical support; as well as all the participants of visual assessment experiments.

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680.

[2] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[3] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *International Conference on Learning Representations*, 2019.

[4] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *CVPR*, 2017, pp. 105–114.

[5] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *CVPR*, 2018.

[6] H. Cai, C. Bai, Y.-W. Tai, and C.-K. Tang, "Deep video generation, prediction and completion of human action sequences," in *The European Conference on Computer Vision (ECCV)*, 2018.

[7] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro, "Video-to-video synthesis," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[8] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds., 2016, pp. 4565–4573.

[9] Y. Li, J. Song, and S. Ermon, "Inferring the latent structure of human decision-making from raw visual inputs," *ArXiv*, 2017.

[10] M. Jegorova, S. Doncieux, and T. M. Hospedales, "Generative adversarial policy networks for behavioural repertoire," in *IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, 2019, pp. 320–326.

[11] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *International Conference on Computer Vision (ICCV)*, pp. 2242–2251, 2017.

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2016.

[13] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[14] C. Esteban, S. Hyland, and G. Rtsch, "Real-valued (medical) time series generation with recurrent conditional gans," *arXiv*, 2017.

[15] A. I. Karjalainen, R. Mitchell, and J. Vazquez, "Training and validation of automatic target recognition systems using generative adversarial networks," *Sensor Signal Processing for Defence*, 2019.

[16] C. Fabbri, M. J. Islam, and J. Sattar, "Enhancing underwater imagery using generative adversarial networks," in *ICRA*, 2018, pp. 7159–7165.

[17] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, p. 66296640.

[18] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv*, 2014.

[19] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollr, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 318–327, 2020.

[20] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *4th International Conference on Learning Representations (ICLR)*, 2016.